

adinstbpm - Task #20437

Milestone # 20350 (New): IOTA BPM deployment

Functional specification document

07/24/2018 04:13 PM - John Diamond

Status:	Assigned	Start date:	07/24/2018
Priority:	Normal	Due date:	
Assignee:	John Diamond	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Description			
Collect functional specification from IOTA specialists and document in the Wiki.			

History

#1 - 07/24/2018 04:14 PM - John Diamond

Here are some specs that I received via e-mail from Chip -

I realized that I forgot to send along the naming convention. This is attached, but the general convention for IOTA BPMs is:

N:IBggyx

where gg is the girder, y is L/R/C (left, right, or center), and x is a single character for channel. Unfortunately this prohibits the use of 2-character designators like we've used with the line, but we really would like to stick with the 8-character legacy naming space for the short character designation. What I would propose is the following for 'x':

H = Horizontal position

V = Vertical position

I = Intensity (amplitude)

A/B/C/D = individual button magnitudes

Between the meeting and speaking with Dan, it sounds like we want [0] for each of these to be sampled periodically (15 Hz to allow for the max datapool update?). [1:~100] should be before beam is injected (this can float a little so if one injection gives us 105 empty buckets before injection and the next gives us 99, that's alright as long as we have at least a reasonable sample of the noise floor ahead of injection according to Sasha R. This may only really mean something for the A/B/C/D magnitudes ahead of fitting, but we probably want to be able to correspond the turns without having to remember an offset between them and the fit H/V/I devices. TBT data should fill the buffer following this ([~100:~8000]).

Other parameters can probably be fit into the namespace as needed with remaining alpha-numeric characters, but if you want guidance on any in particular, I can make suggestions.

Naming for the full orbit devices should be N:IBPMx, where x again is H/V/I. Elements for these devices should be in order around the ring from injection ([0:28]): A1C, A2R, A3R, R1R, B1R, B2R, R2R, C1R, C2R, R3R, D1R, D2R, R4R, E1R, E2R, E2L, E1L, R4L, D2L, D1L, R3L, C2L, C1L, R2L, B2L, B1L, R1L, A3L, A2L. While this would be for the periodic BPM sample ([0] for N:BgyH/V/I), it would be handy for our synoptic displays to be able to tell the frontend to load a given TBT frame too. I would propose this to be in a second set of orbit devices N:IBPMTx (x=H/V/I/N, where N would just be the index of the element set you want copied into the H/V/I devices).

Taking all this into account, the BPMs would work as follows:

An event is generated to arm the kickers/BPMs/etc and the BPMs start the TBT buffer. Each BPM device (e.g. N:BA1CH, or the BPM on girder A1, center position on the ring between the left/right symmetry, horizontal position) will be recording the periodic sample of the position into element [0], ~100 elements prior to injection into [1:100], and then all TBT information into the elements after that to fill out the array. The full orbit device will be updated with all the [0] elements for all the BPMs around the ring (e.g. N:IBPMH⁰ = N:IBA1CH⁰, N:IBPMH¹ = N:IBA2RH⁰, N:IBPMH² = N:IBA3RH⁰, ... , N:IBPMH²⁸ = N:IBA2LH⁰). If the operator wants to inspect the orbit at turn ~1000 in the BPM synoptic display, they'll set N:IBPMTN to 1100 on the synoptic display and the BPM front end should take elements [1100] for each of the BPM devices and build up the tbt orbit device (e.g. N:IBPMTH⁰ = N:IBA1CH¹¹⁰⁰, N:IBPMTH¹ = N:IBA2RH¹¹⁰⁰, N:IBPMTH² = N:IBA3RH¹¹⁰⁰, ... , N:IBPMTH²⁸ = N:IBA2LH¹¹⁰⁰).